Linear Regression
Bias Variance Tradeoff

BA810: Supervised Machine Learning

Nachiketa Sahoo

Recap

- What is machine learning?
  - A program that improves at a given task with increased experience
- Different types of learning
  - Supervised vs unsupervised
  - Regression vs classification
  - Prediction vs inference
- Tradeoff between flexibility and interpretability of predictive models

- Linear regression
  - $y \ = \ f(x) = \beta x + e$
  - Residuals $r_i = y_i - \hat{y}_i$
  - Mean Squared Error (MSE): average of squared residuals
  - Linear regression minimizes MSE in the training sample
- Train-test paradigm to measure error out of sample
  - Split the data into two parts; use one for training and the other for testing

Outline

- Interpreting linear regression results
- Overfitting and underfitting
- The bias-variance tradeoff in machine learning

Assessing the Fit of the Model

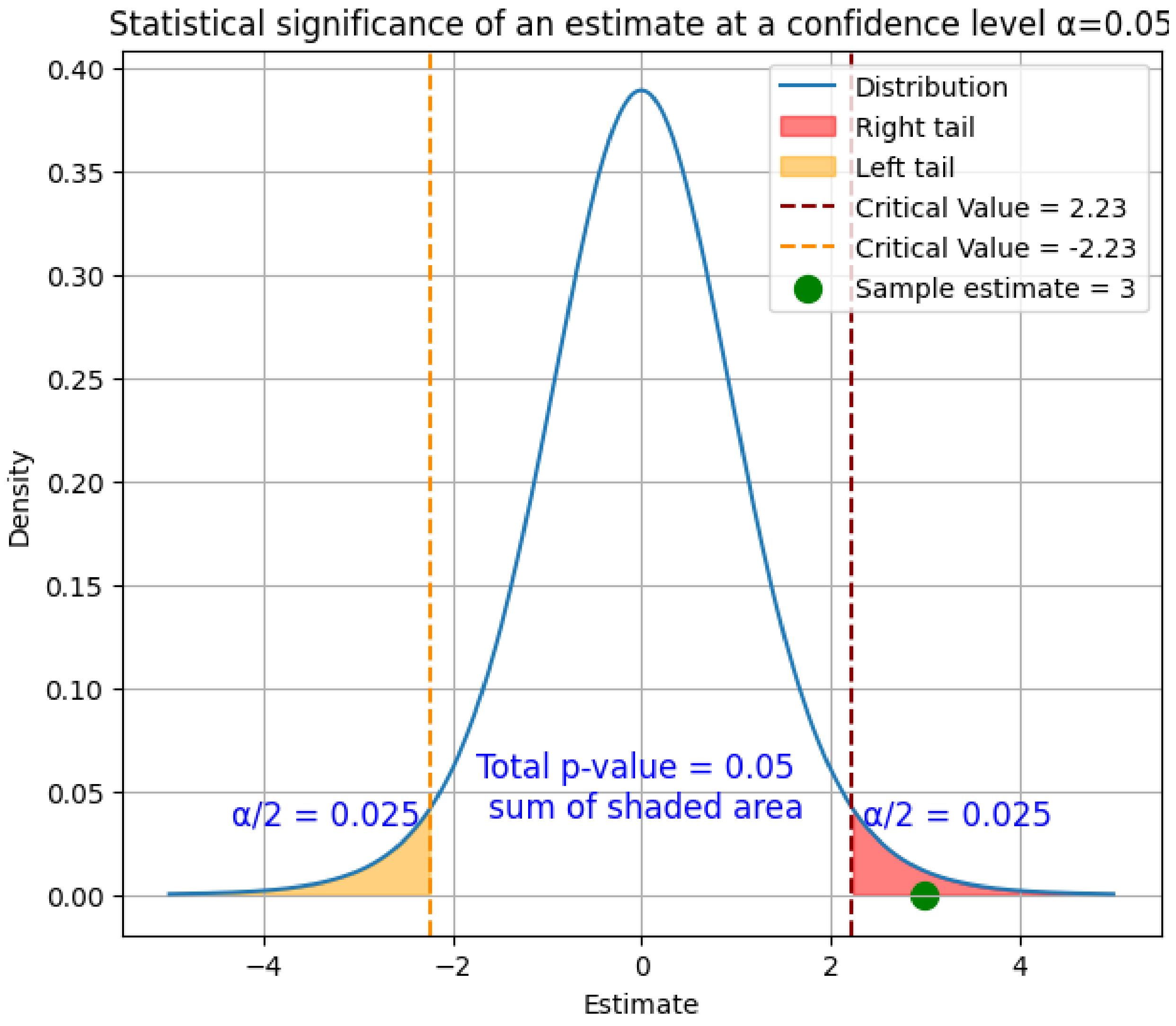
- Given linear regression estimated model
$$\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x$$
- Measure of error: $\sum_i (y_i - \hat{y}_i)^2$, aka residual sum of squares (RSS)
- $TSS = \sum_i^n (y_i - \bar{y})^2$: Total Sum of Squares (error of the null-model)
- Fit of the model to the data:
$$R^2 = 1 - \frac{RSS}{TSS}$$

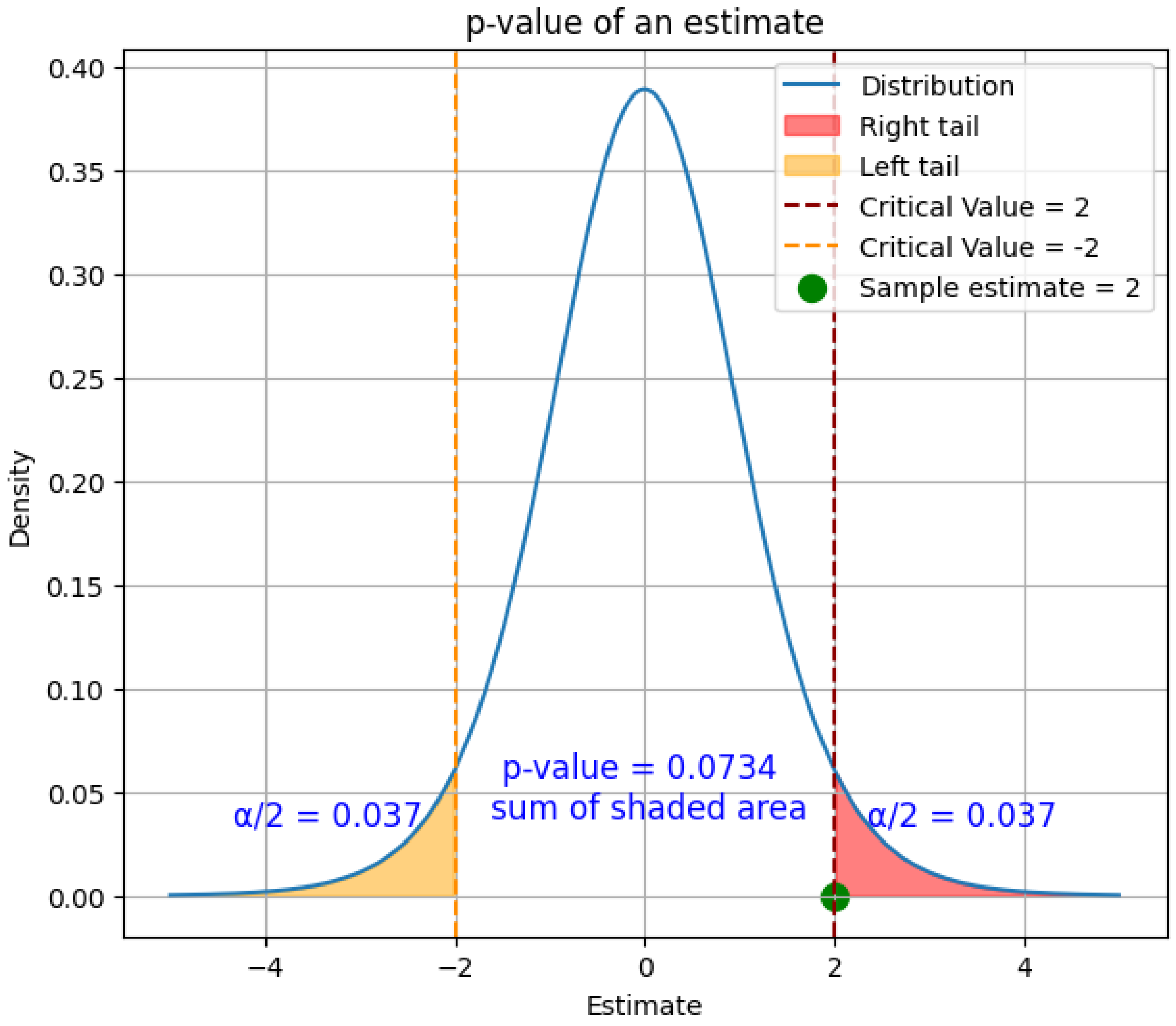
R^2: the fraction of the variation in y explained by our fitted model

Interpreting the Coefficients

- $\hat{\beta}_1$: how much y increases when x increases by 1.
- How reliable is the estimated coefficient?
  - Use standard deviation of the $\hat{\beta}_1$s under repeated sampling
  - Can be approximated from the one data sample used to fit the regression
- The 95% confidence interval $\approx \hat{\beta}_1 \pm 2 \times \mathrm{SE}(\hat{\beta}_1)$
  - Includes 0 → no effect
- p-value:
  - Probability that an estimated coefficient is "as extreme as" $\hat{\beta}_1$ by random chance *even when the true β_1 was 0!*
  - Unlikely if p-value < 0.05 or 0.01; likely that there was a non-zero effect

Interpreting the Coefficients





Multiple linear regression

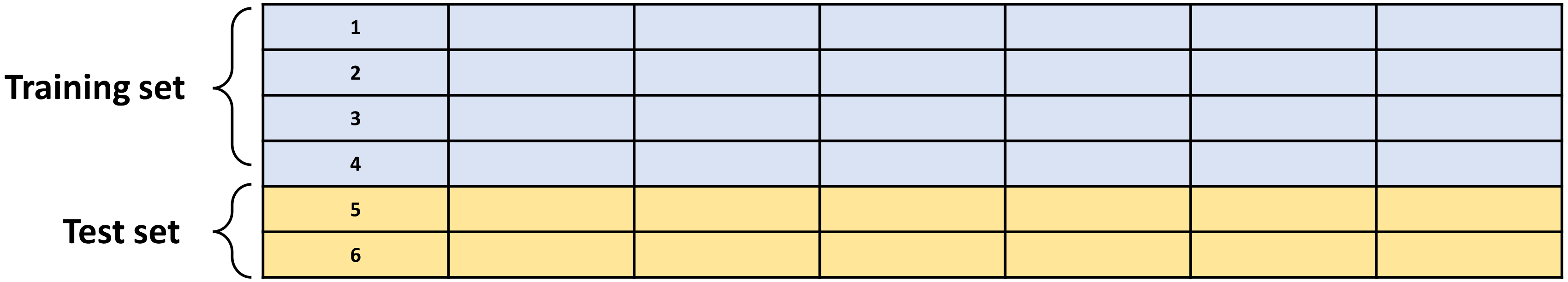
- Linear regression with more than one explanatory variables

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

  β_2 is how much y increases when x_2 increases by 1 *holding* x_1 *fixed*

- Note: “linear” refers to linearity of *parameters*
  - Using polynomial features — $x^2, x^3, \ldots$ — in a linear regression one can fit highly non-linear lines to data
- Which of the following is a linear regression?
  a. $y = \beta_0 + \beta_1 x_2 \log(x_1) + \beta_2 e^{x_2} + \epsilon$
  b. $y = \beta_0 + \beta_1 x_1 + \log(\beta_2)\, x_2 + \epsilon$

Overfitting
Underfitting



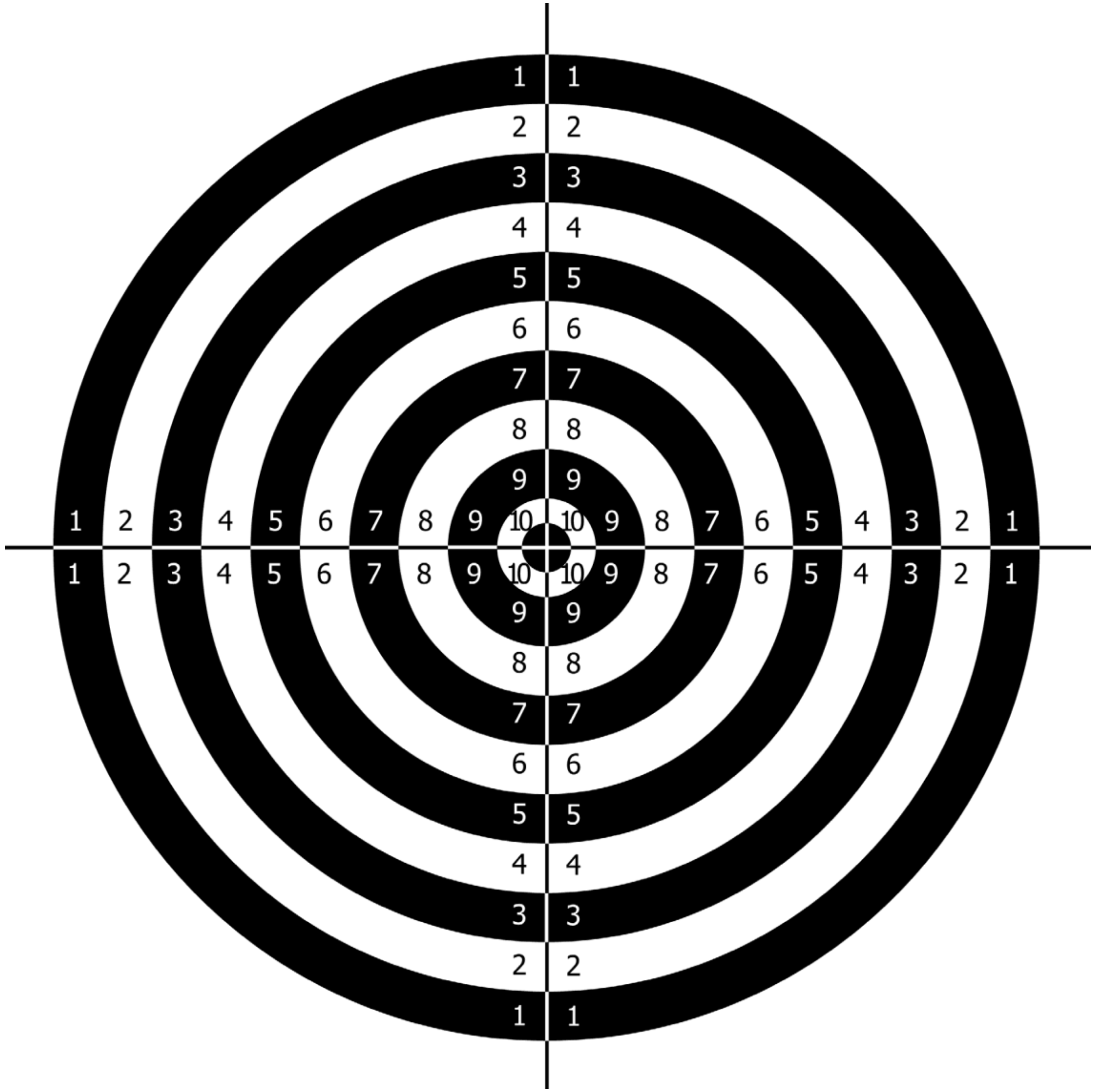
- Fit a linear regression to training set, then measure the error on training (MSE_{train}) and test data (MSE_{test})
- Which one do you expect to be larger? Why?

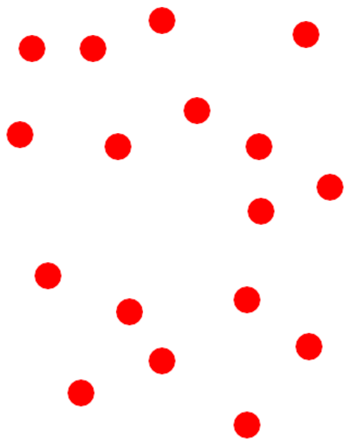
$$MSE_{train} < MSE_{test}$$

Overfitting	Underfitting
• MSE_{train} is small, but MSE_{test} is large • Potential reasons: model too complex/flexible or too little data	• Both MSE_{train} and MSE_{test} are large • E.g., close to that of null model • Potential reasons: model too simple, training incomplete

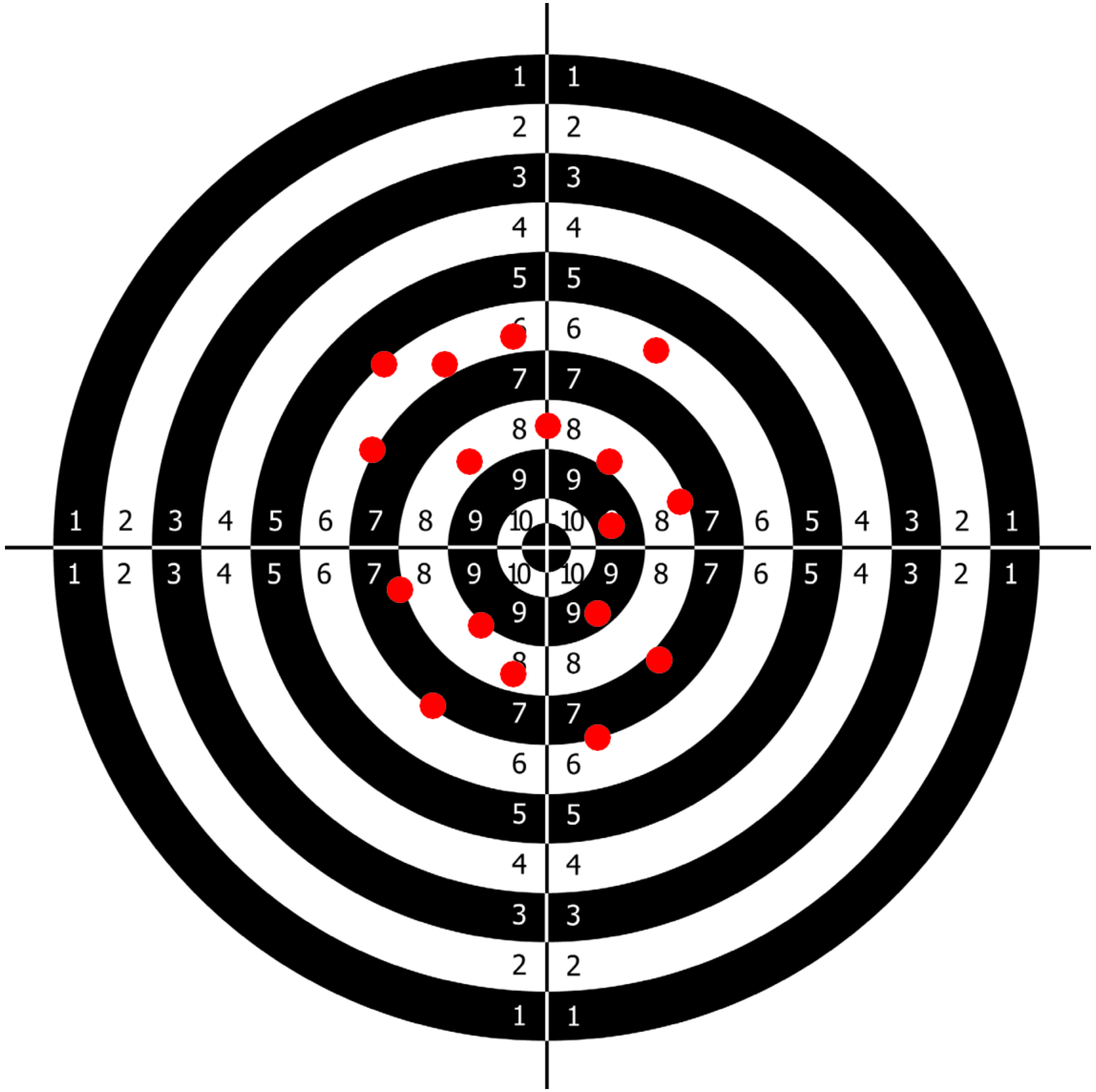
The Bias-variance Tradeoff

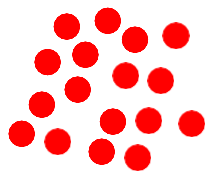
- Reducing MSE on your training data does not necessarily reduce MSE on data you have not trained on
  - MSE_{test} could be going up as MSE_{train} is going down!
  - Because of a fundamental trade-off in machine learning, called the bias–variance trade-off
- What is bias? What is variance?
  - Note: prediction at a point changes with training data
  - Let's get some intuition…


1 2 3 4 5 6 7 8 9 10 10 9 8 7 6 5 4 3 2 1
1 2 3 4 5 6 7 8 9 10 10 9 8 7 6 5 4 3 2 1

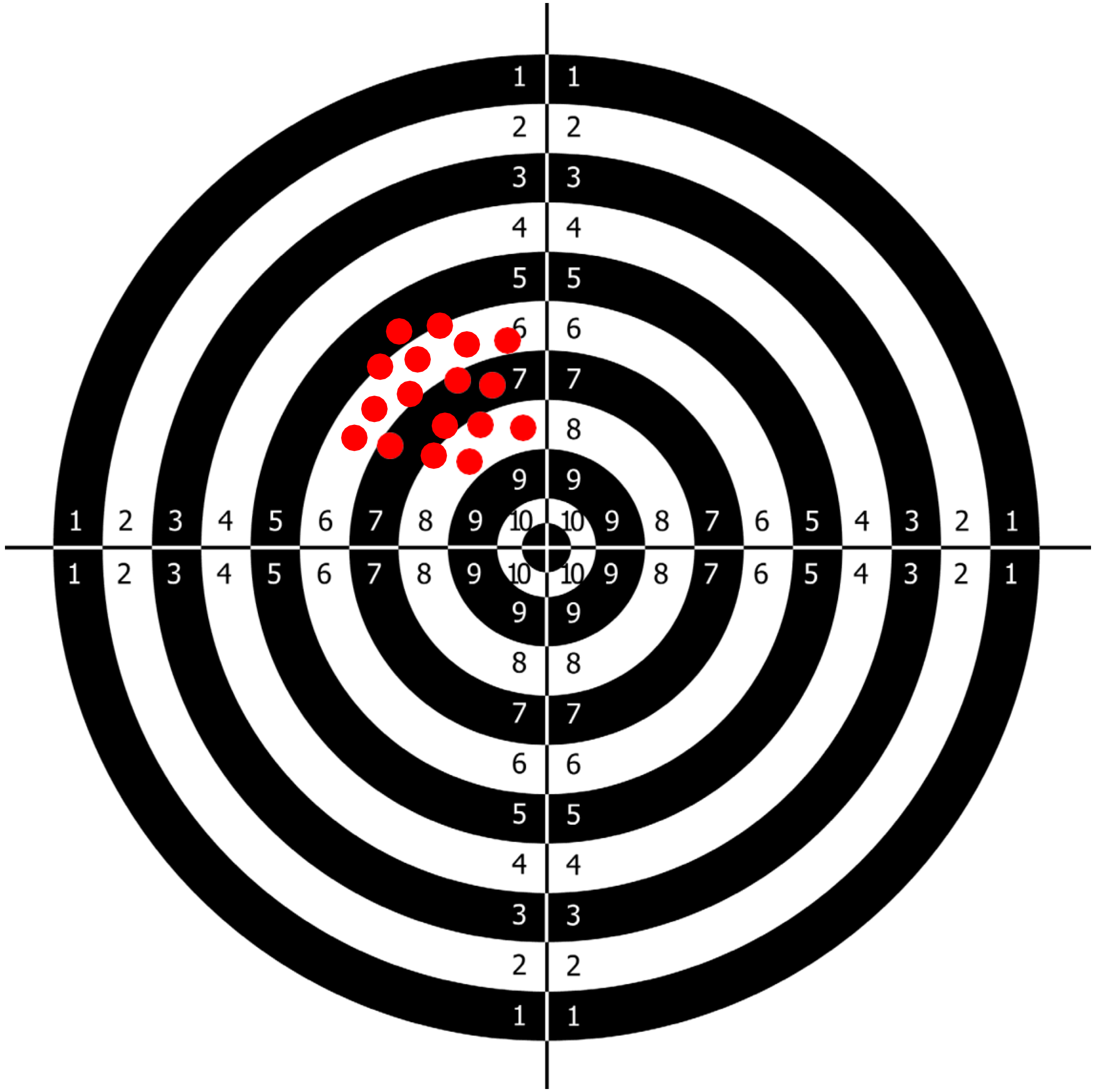


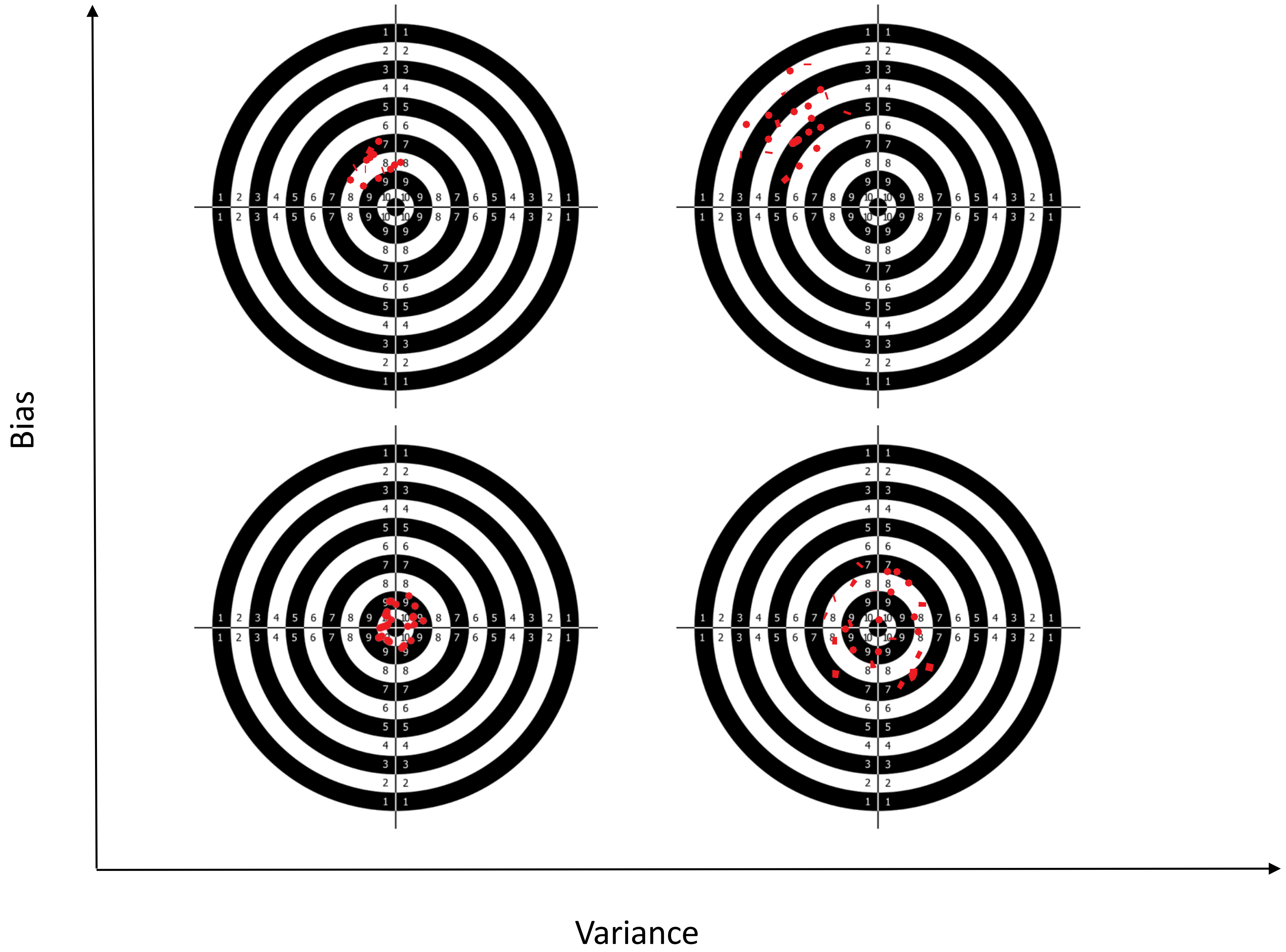
High variance



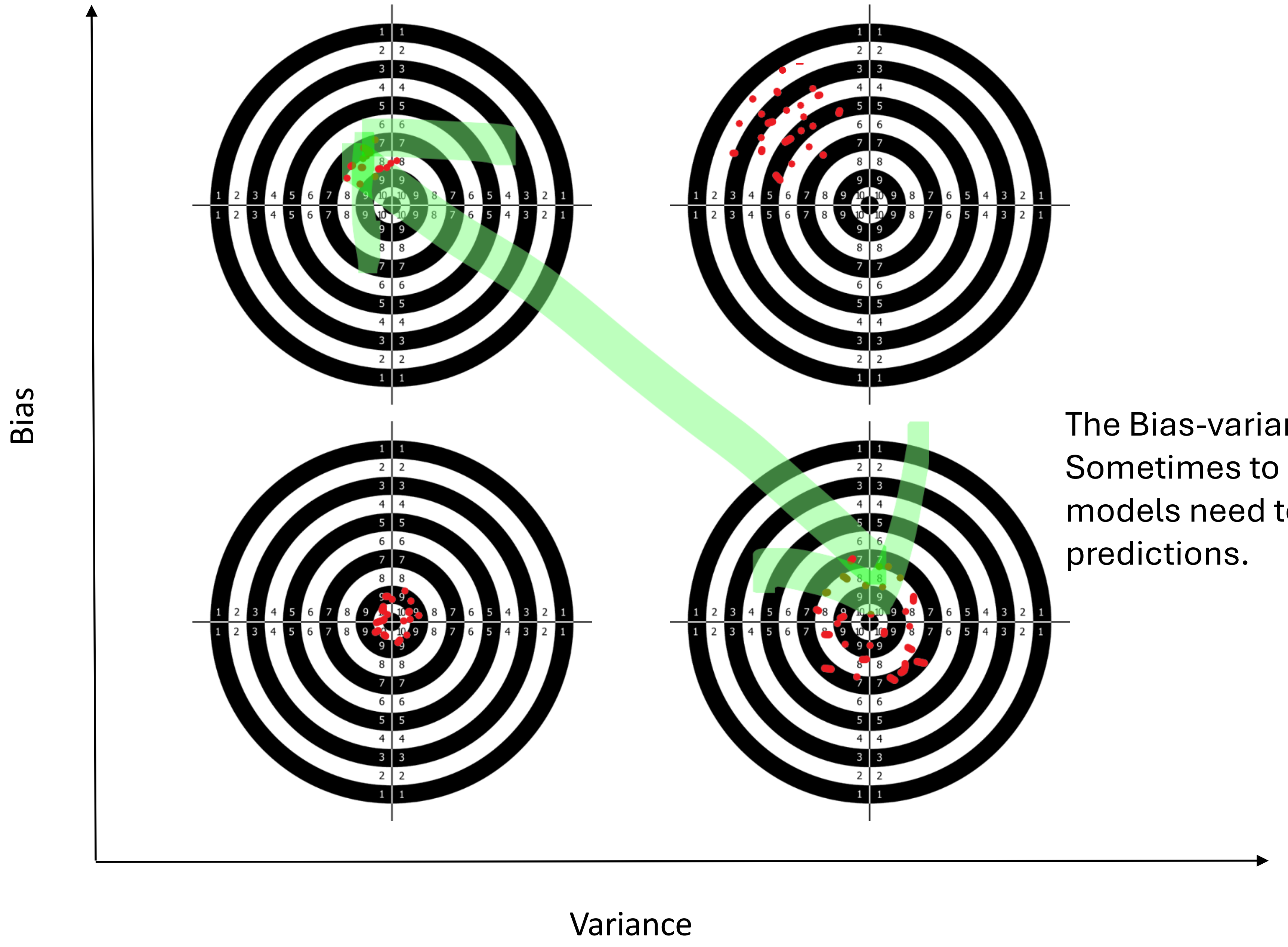


High bias





Bias
Variance



The Bias-variance trade-off: Sometimes to reduce error models need to bias the predictions.

A little more formal definition

- Suppose we create many datasets from the same source
  - E.g., by randomly sampling a subset from a larger dataset
- For each dataset indexed by $b = 1 \dots B$ we estimate a model $\hat{f}^{(b)}$ with parameters $\left(\hat{\beta}^{(b)}\right)$

Then for a point x_0 we predict $\hat{y}^{(b)} = \hat{f}^{(b)}(x_0)$

- **Bias:** gap between the average prediction $\frac{\sum_b \hat{y}^{(b)}}{B}$ (over B training sets) and the true value y_0.
- **Variance:** variance of predictions at $\left\{\hat{f}^{b}(x_0)\right\}$ as we learn from different samples.

Bias-variance trade-off

- Let $Y = f(x) + e$
- The average prediction error can be written as:

$$MSE = E\left(\left(Y - \hat{f}(x)\right)^2\right) = \underbrace{Var\left(\hat{f}(x)\right) + Bias\left(\hat{f}(x)\right)^2}_{\text{Reducible Error}} + \underbrace{Var(e)}_{\text{Irreducible Error}}$$

where $Bias\left(\hat{f}(x)\right) = f(x) - E\left(\hat{f}(x)\right)$

As models become more flexible (bias reduces) they become more sensitive to change in data (variance increases).

Illustration

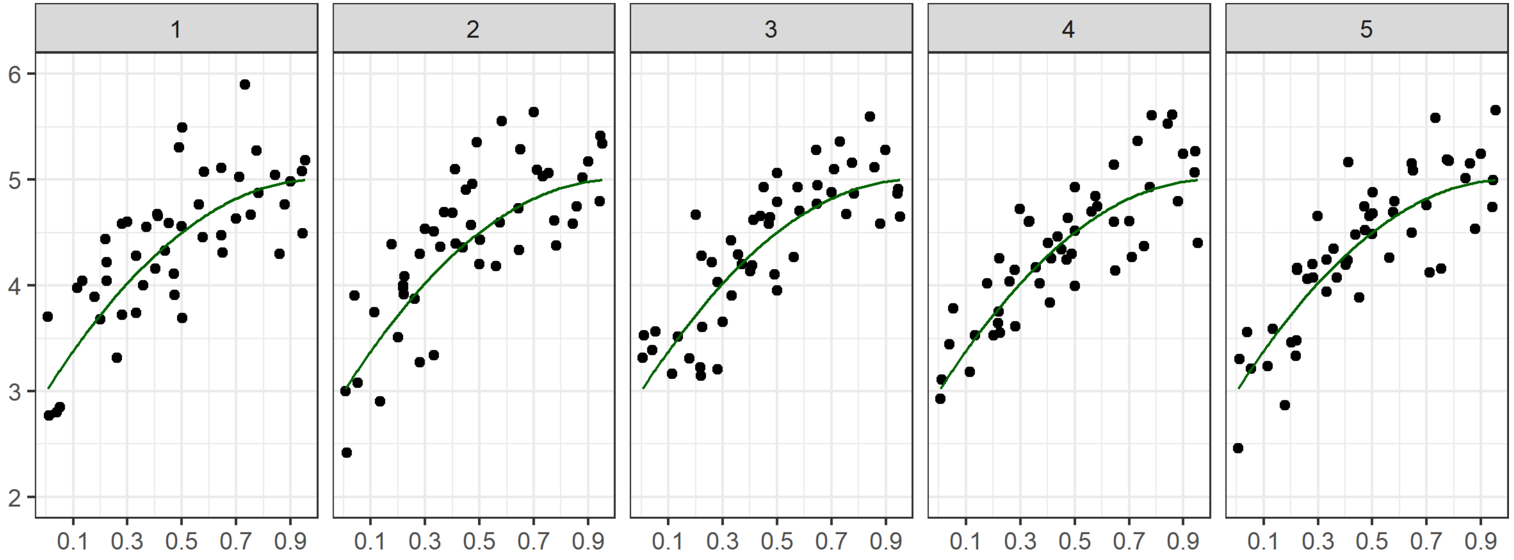
- True relationship

$$Sales(AdSpend) = 3 + 4 \times AdSpend - 2 \times AdSpend^2$$

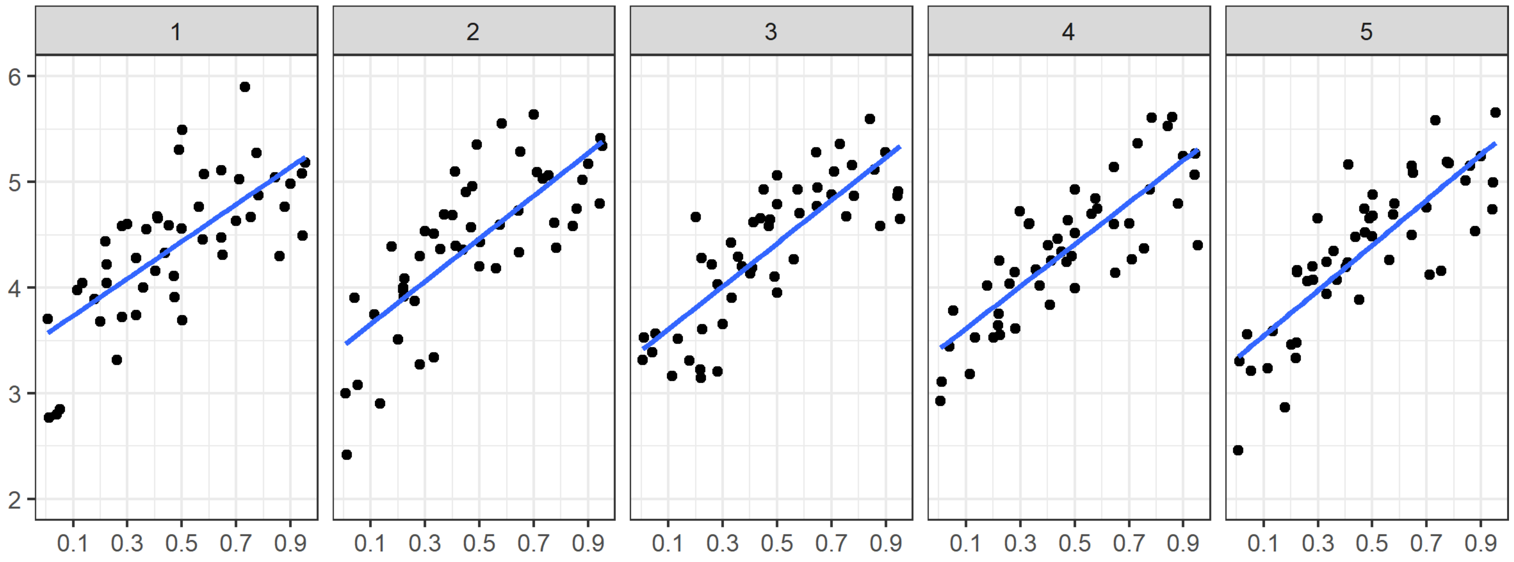
but some random noise is added at observation

- Draw five different datasets, i.e., our data are from

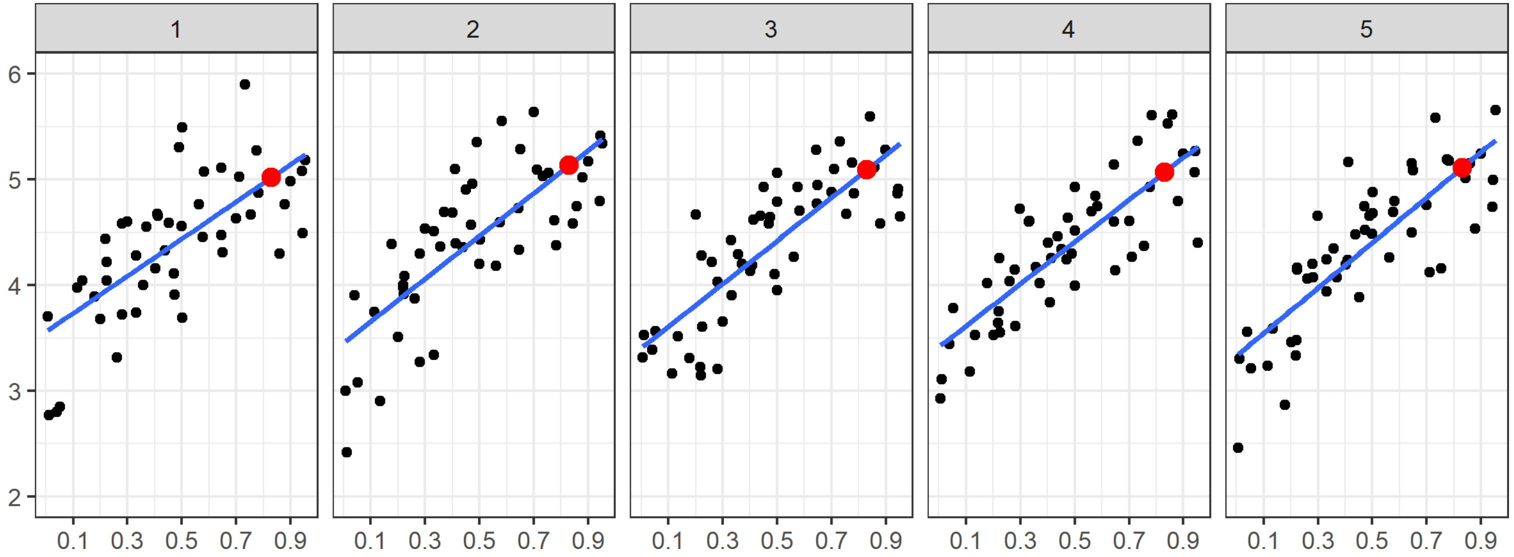
$$y = Sales(AdSpend) + e$$



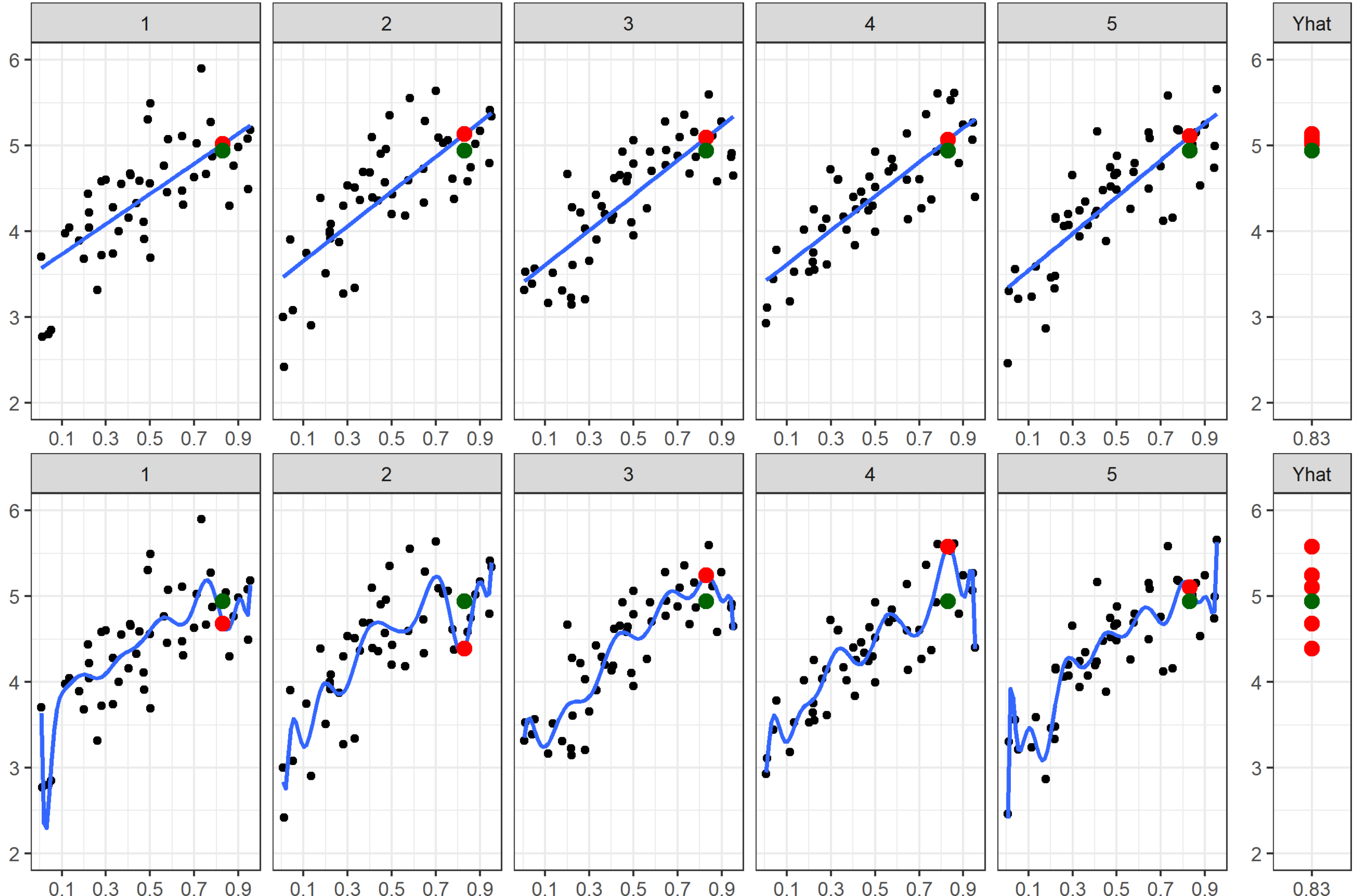
Fit a linear regression

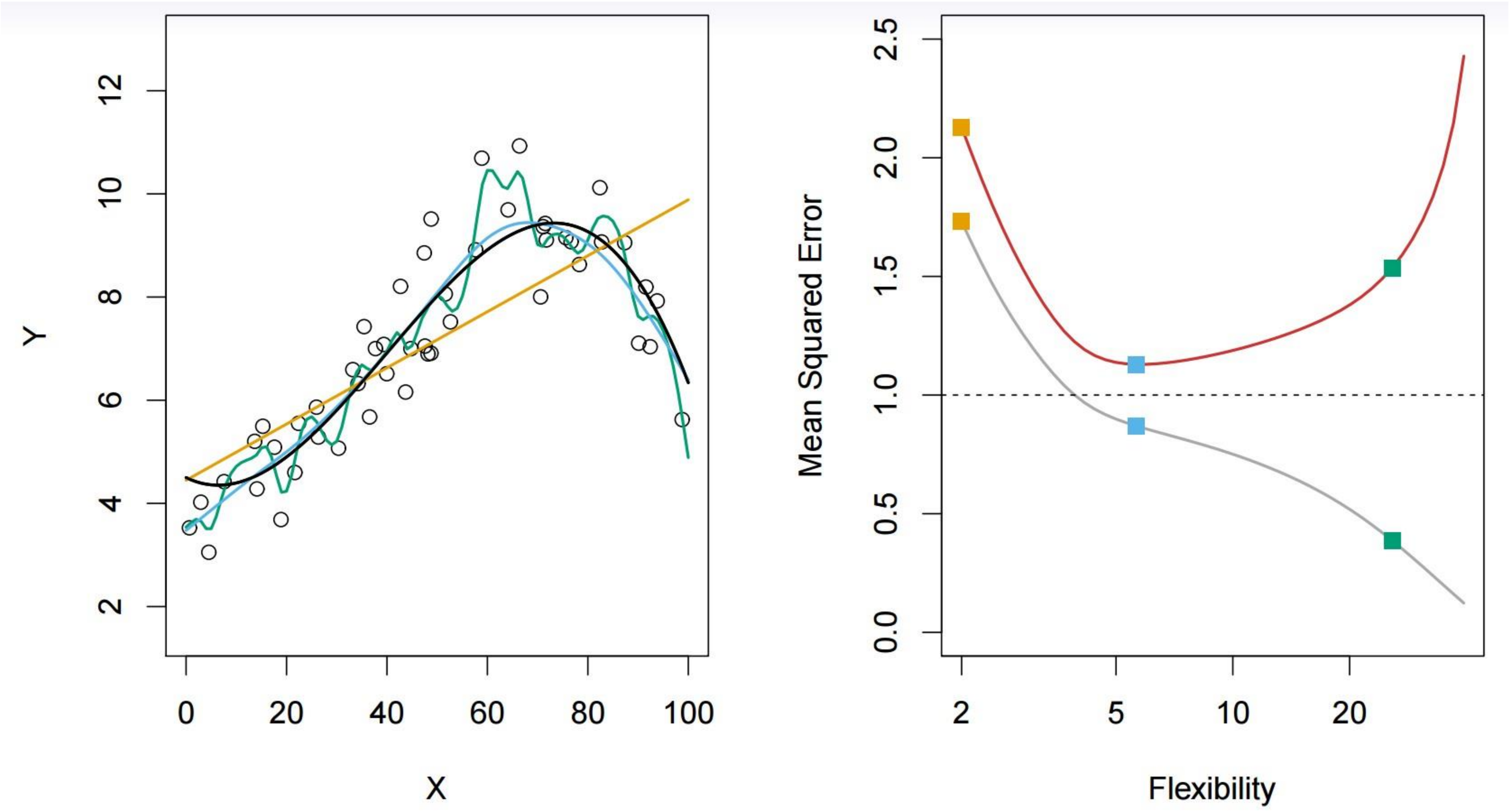


Predict y at a point that truly is at the red dot



Predict y at a point that truly is at the green dot





Black curve is truth. Red curve on right is MSE_{Te}, grey curve is MSE_{Tr}. Orange, blue and green curves/squares correspond to fits of different flexibility.

Navigating the bias-variance trade-off

- How do we balance model complexity and prediction accuracy?
- Train models with various degrees of complexity, watch test data error
  - Pick the model with the lowest error
- Could work when the number of models to consider are small (e.g., polynomials of degree 1 … 20)
- The options can grow exponentially in some cases (e.g., which of the 20 features to include?)
  - Need smarter heuristics (to be discussed in a later class)

Summary

- R^2: fraction of variation in Y explained by the regression
- β_k: how much y increases when x_k increases by 1, holding other xs constant
- p-value of $\hat{\beta}_k$: probability of obtaining as extreme a value by chance even if true β_k was 0
- Overfitting: fitting training data too closely to predict well on test data
  - Simplify model, get more data
- Underfitting: fitting both training and test data poorly
  - Use more flexible model

Summary

- Bias: if we refit the model to different samples, how different is the average prediction from the truth
- Variance: how dispersed are the model predictions for the same point
- Simpler models have higher bias, but lower variance
  - Consider predicting average all the time
- Complex models trade bias for variance